

Fermilab

Particle Physics/Electrical Engineering Department

**Specification for the CMS Hadron Calorimetry
Front End Readout Module
Channel Control ASIC**

Theresa Shaw,
Alan Baumbaugh, Ahmed Boubekeur, John Elias, Ray Yarema

November 23, 1999
Revised
January 23, 2004

Revision History:

09/14/2000	TMS	Fixed Fig.1 and Fig. 2 to show clocking coming from Master Clock source and not CCA. Fixed Data Format and Orbit Message format to match up with CHFET Serializer specification. Added “Send Fill Frame” bit to Control Register.
09/26/2000	TMS	Simplify CAP ID checking – no more voting logic. Just flag CAP ID mismatch and flag error. Added detail. Added pinout table. Added simulation ideas.
10/30/2000	TMS	Changed: Serial bus named RBXbus. Worst case channel to channel spread changed from 15ns to 58ns. Appendix A – this logic suggested by Al Baumbaugh for the alignment of the data from the two QIEs serviced by the CCA. Added some pin info – we now have 85 signal pins Added Beam Zero timing info Added second QIE_RESET pulse, so that each QIE channel has its own - implementation of QIE_Reset is described in Appendix A Added an RBXbus register to control alignment mux's described in Appendix A Test Pulses changed from 10 clock cycle width to 1 clock cycle width AND polarity control. Also, these pulses must be timed in the same way as the QIE_RESET pulses, see APPENDIX A. Orbit Message is now sent upon the receipt of Beam_zero. Previously, it worked off a counter.
1/9/2001	TMS	Redo Serializer Output section to match CERN GOL ASIC Both CAP IDs will be drive out of CCA RBXbus Address pins RBX_A(7:2) all programmable Polarity change of Send_Fill_Frame signal Add fill frames to Orbit message
7/12/2001	TMS	Add pinout information
2/12/2002	TMS	Revise pinout information; new package 120 pin 14mmx20mm; rename some of “die” pad names to make PCB routing easier. For instance, Mant 0 thru 4, renamed Mant 4 thru 0. This changes definition of TXD output bits.
2/19/2002	TMS	revise pinout to new package 128pin 14mm x 20mm
3/08/2002	TMS	revise pinout to add package pin for leadframe die pad
10/2/2002	TMS	general edits
1/23/2004	TMS	Fixed bit position definition of QIE_Force_Range(1:0). The bits were reversed in prior documentation

Introduction

The CMS Hadron Calorimetry Front End Readout Module is based upon the specification, design and production of several ASIC devices.

The FE Module will receive inputs from six Hybrid Photodiodes (HPDs). Each of these inputs will be fed into a “QIE” charge integrating ASIC which has a built in Flash ADC. The digital data resulting from the conversion is then passed to the Channel Control ASIC.

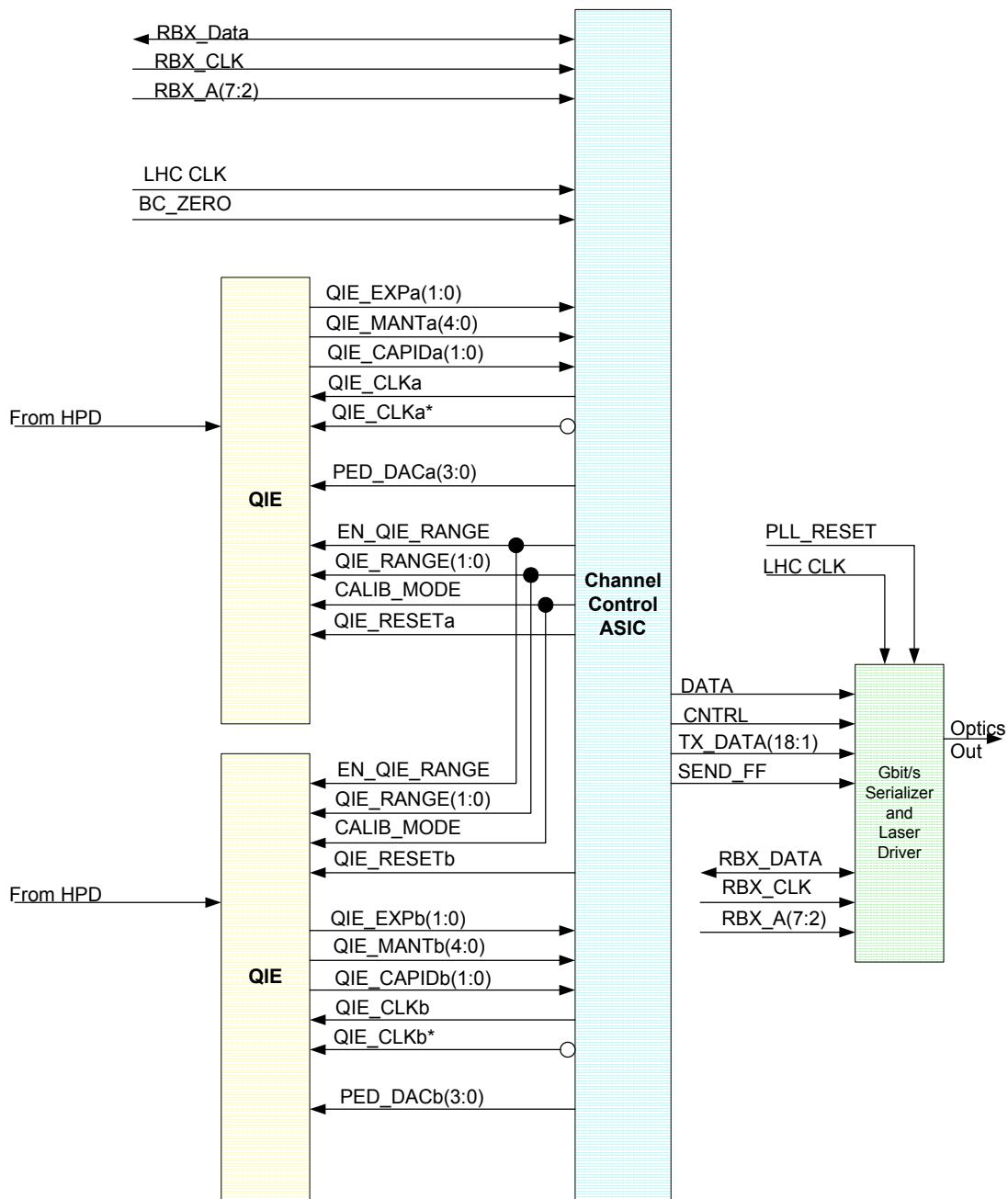
Figure 1 illustrates the connections between the Channel Control ASIC and other FE module components. Figure 2 is a block diagram of the Channel Control ASIC.

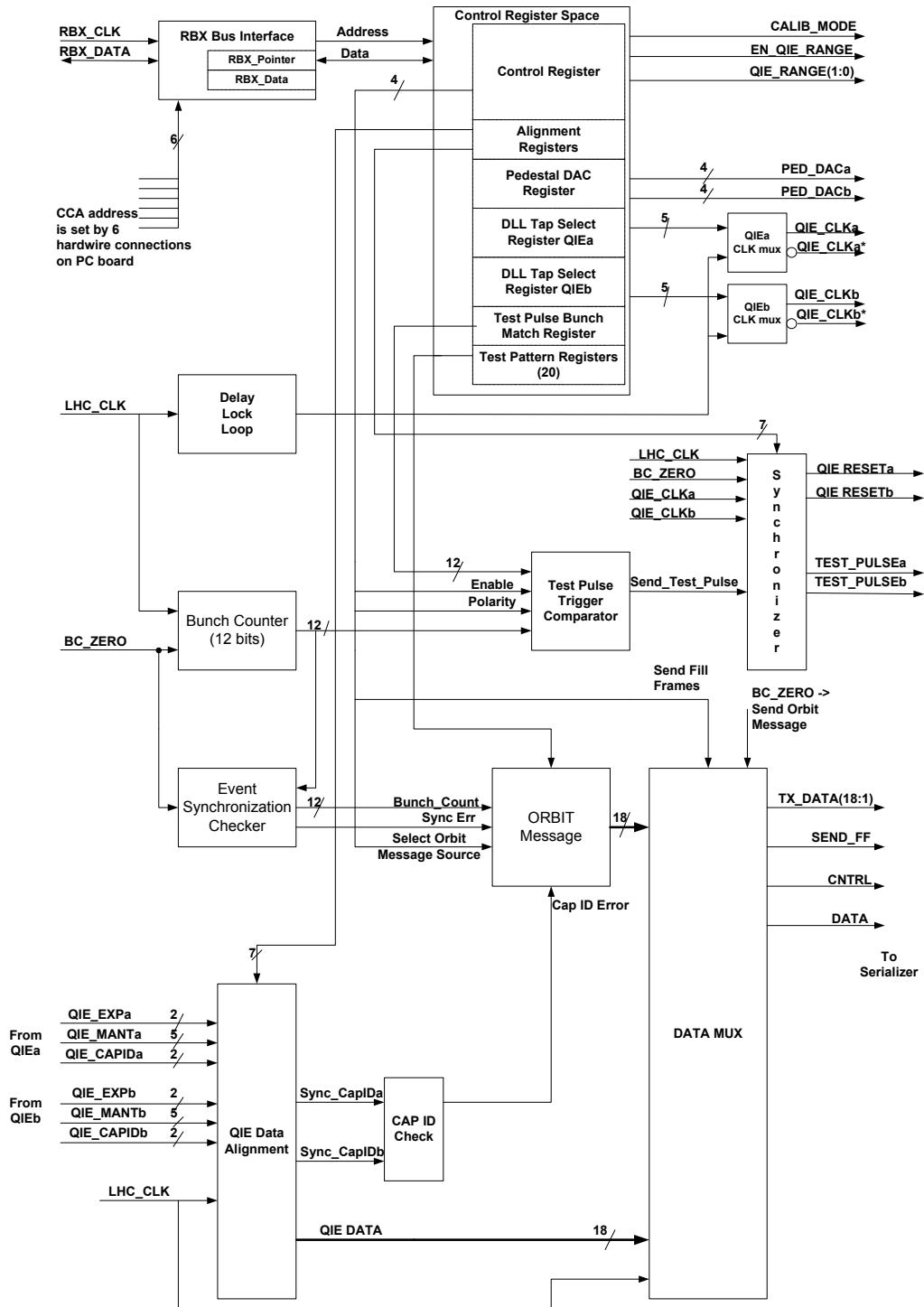
Each Channel Control ASIC (CCA) receives the digital data from two QIEs. Each QIE presents a 5 bit mantissa, a 2 bit exponent and a 2 bit Capacitor ID.

The CCA must provide the following functions:

- The processing and synchronization of the data from two QIEs,
- The provision of QIE clocking signals to run the QIE charge integrator and Flash ADC,
- Checking of the accuracy of the Capacitor IDs (the Cap IDs from different QIEs should be in synchronization),
- The ability to force the QIE to use a given range,
- The ability to set QIE Pedestal DAC values,
- The ability to issue two test pulse triggers of programmable polarity,
- The provision of event synchronization checks – a crossing counter will be implemented and checked for accuracy with every beam turn marker,
- The ability to send a known pattern to the serial optic link,
- The ability to “reset” the QIE,
- And, the ability to send and report on any detected errors at a known and determined time.

A serial link will be implemented on the Channel Control ASIC to allow users to download information which controls its operation.

**Figure 1** Channel Control ASIC Connections

**Figure 2 Channel Control ASIC Block Diagram**

The RBXbus and Register Access

The RBXbus protocol will be used to communicate with the Channel Control ASIC. The RBXbus is an asynchronous 2-wire serial bus and allows for read/write operations at up to 100 Kbits/s. RBXbus is similar to the well-known I²C protocol bus. Devices following standard I²C protocol will be able to communicate with the CCA.

Data Transfer via the RBXbus

The RBXbus provides a seven bit address field. To provide communication with the CCA, each CCA will implement two RBXbus registers. The first register will be an address pointer, **RBX_Pointer**. This will be an eight bit register which is loaded with an address pointer to register space.

The second RBXbus register implemented by each CCA will be an eight bit **RBX_Data** register. The data written to this register will actually be written to the register space which is pointed to by **RBX_Pointer**. Likewise, data read from the **RBX_Data** register will come from the register being pointed to by **RBX_Pointer**.

Thus; to accomplish communication with the CCA, only two RBXbus registers must be addressable on each device. Six “ID” or programmable address pins, **RBX_A(7:2)**, will be used to set the individual CCA addresses. These may be hardwired on the printed circuit board. **RBX_A1** will be used to determine whether the **RBX_Pointer** or the **RBX_Data** register is being addressed.

The RBXbus addresses for the two registers will be set as follows:

CCA RBXbus address space (7 bit address field)

RBXbus Register	A7	A6	A5	A4	A3	A2	A1
RBX_Pointer	RBX_A7	RBX_A6	RBX_A5	RBX_A4	RBX_A3	RBX_A2	0
RBX_Data	RBX_A7	RBX_A6	RBX_A5	RBX_A4	RBX_A3	RBX_A2	1

Register Space

The following registers are read and written through the RBX serial bus interface. Default power on value should be “low” in all cases.

Control Register

General Description: The Control Register contains bits to turn on/off various CCA functions.

RBXbus Pointer Address = 0x00

<u>Bit</u>	<u>Definition</u>
7	Polarity of Test Pulse 0 – Test Pulse will be high going pulse 1 – Test Pulse will be low going pulse
6	Send Fill Frames <i>This bit, when set, drives the output signal “Send_Fill_Frames” high.</i>
5	Enable Calibration Mode
4	QIE Force Range(0)
3	QIE Force Range(1)
2	Enable QIE Force Range <i>This bit will cause the QIE to stop autoranging, and use the range indicated though the QIE Force Range(1:0) bits.</i>
1	Select Orbit Data <i>This bit controls which of the two sources is selected for the Orbit Message.</i> 0 – Orbit message will consist of one word of Cap ID synchronization checking, one word of Bunch Counter synchronization checking and 80 words of fill frames. 1 – Orbit message will be the 20 bytes of Test Pattern Data packed as 16-bit words.
0	Enable Test Pulse <i>This bit will enable the sending of a Test Pulse when matches occur with the Test Pulse Bunch Match Register and the Beam Counter.</i>

Alignment Control QIEa Register

General Description: The Alignment Control QIEa Register contains bits to control the QIEa channel alignment features.

Refer to Appendix A for explanation of these settings.

RBXbus Pointer Address = 0x01

<u>Bit</u>	<u>Definition</u>
7	Undefined
6:5	Selects phase of QIEa Data <i>(MUX D in Appendix A)</i> 0 – selects Q0D2 1 – selects Q0D3 2 – selects Q0D4 3 – undefined
4	LHC_CLK polarity select – for data alignment of QIEa <i>(Mux C in Appendix A)</i> 0 – selects inverse of LHC_CLK 1 – selects LHC_CLK
3	Undefined
2:1	Selects phase of QIE_RESETa or TEST_PULSEa signal <i>(MUX B setting in Appendix A)</i> 0 – selects Q0R2 1 – selects Q0R3 2 – selects Q0R4 3 – undefined
0	QIE_CLKa polarity select for aligning QIE control signal QIE_RESETa or TEST_PULSEa <i>(MUX A setting in Appendix A)</i> 0 – selects inverse of QIE_CLKa 1 – selects QIE_CLKa

Alignment Control QIEb Register

General Description: The Alignment Control QIEb Register contains bits to control the QIEb channel alignment features.

Refer to Appendix A for explanation of these settings.

RBXbus Pointer Address = 0x02

<u>Bit</u>	<u>Definition</u>
7	Undefined
6:5	Selects phase of QIEb Data <i>(MUX D in Appendix A)</i> 0 – selects Q0D2 1 – selects Q0D3 2 – selects Q0D4 3 – undefined
5	LHC_CLK polarity select – for data alignment of QIEb <i>(Mux C in Appendix A)</i> 0 – selects inverse of LHC_CLK 1 – selects LHC_CLK
3	Undefined
2:1	Selects phase of QIE_RESETb or TEST_PULSEb signal <i>(MUX B setting in Appendix A)</i> 0 – selects Q0R2 1 – selects Q0R3 2 – selects Q0R4 3 – undefined
0	QIE_CLKa polarity select for aligning QIE control signal QIE_RESETb or TEST_PULSEb <i>(MUX A setting in Appendix A)</i> 0 – selects inverse of QIE_CLKb 1 – selects QIE_CLKb

Pedestal DAC Register

General Description: The Pedestal DAC Register contains the settings of two 4-bit QIE Pedestal DACs.

RBXbus Pointer Address = 0x03

<u>Bit</u>	<u>Definition</u>
(7:4)	PED_DACb(3:0)
(3:0)	PED_DACa(3:0)

DLL Tap Select Register QIEa

General Description: The DLL Tap Select Register QIEa contains the select code for the DLL phase chosen to run the QIE_CLKa. The DLL must allow for a 25ns shift of the clock phase in increments of 1ns.

Note: Valid settings are 0x00 through 0x19 (0 through 25ns). Settings higher than 0x19, shuts QIE_CLKa off.

RBXbus Pointer Address = 0x04

<u>Bit</u>	<u>Definition</u>
(7:5)	Undefined
(4:0)	Select (4:0)

DLL Tap Select Register QIEb

General Description: The DLL Tap Select Register QIEb contains the select code for the DLL phase chosen to run the QIE_CLKb. The DLL must allow for a 25ns shift of the clock phase in increments of 1ns.

Note: Valid settings are 0x00 through 0x19 (0 through 25ns). Settings higher than 0x19, shuts QIE_CLKb off.

RBXbus Pointer Address = 0x05

<u>Bit</u>	<u>Definition</u>
(7:5)	Undefined
(4:0)	Select (4:0)

Test Pulse Bunch Match Registers

General Description: The Test Pulse Bunch Match Register contains the Bunch Count at which a Test_Pulse should be fired, providing the “Enable Test Pulse” bit has been set in the Control Register.

RBXbus Pointer Address = 0x07
(MSB)

RBXbus Address = 0x06 (LSB)

<u>Bit</u> (7:4)	<u>Definition</u>	<u>Bit</u> (7:0)	<u>Definition</u>
(3:0) Match(11:8)	Undefined		Match (7:0)

Test Pattern Registers

General Description: This Register Block contains 10 16-bit words which can be the source of the Orbit Message whenever the “Select Orbit Data” bit in the Control Register is set “high”.

RBXBUS Pointer Address = 0x08 thru 0x21

Register Summary

RBXBUS Pointer	Register Name
0x00	Control
0x01	Alignment Control QIEa
0x02	Alignment Control QIEb
0x03	Pedestal DAC
0x04	DLL Tap Select QIEa
0x05	DLL Tap Select QIEb
0x06	Test Pulse Bunch Count Match (LSB)
0x07	Test Pulse Bunch Count Match (MSB)
0x08	Test Pattern Byte 0
0x09	Test Pattern Byte 1
0x0A	Test Pattern Byte 2
0x0B	Test Pattern Byte 3
0x0C	Test Pattern Byte 4
0x0D	Test Pattern Byte 5
0x0E	Test Pattern Byte 6
0x0F	Test Pattern Byte 7
0x10	Test Pattern Byte 8
0x11	Test Pattern Byte 9
0x12	Test Pattern Byte 10
0x13	Test Pattern Byte 11
0x14	Test Pattern Byte 12
0x15	Test Pattern Byte 13
0x16	Test Pattern Byte 14
0x17	Test Pattern Byte 15
0x18	Test Pattern Byte 16
0x19	Test Pattern Byte 17
0x1A	Test Pattern Byte 18
0x1B	Test Pattern Byte 19

DLL

The basic Delay Locked Loop is comprised of 25 one nanosecond delay stages along with a Phase Detector, a Charge Pump, and a Loop Filter as shown in the figure 3 below.

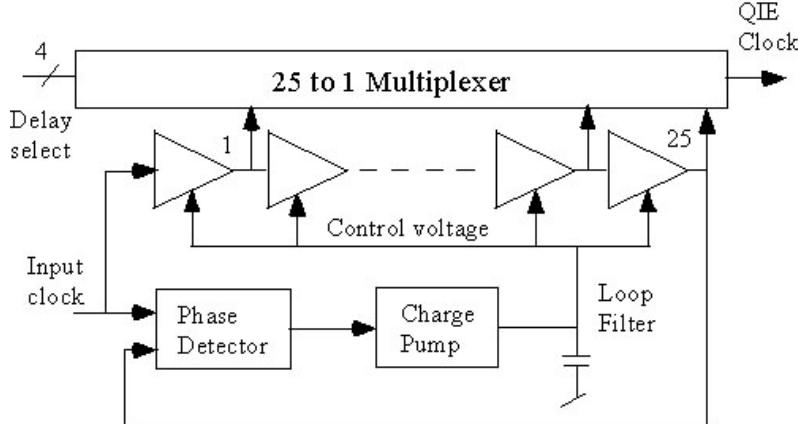


Figure 3. Delay Lock Loop

Each delay stage is comprised of two inverters. Control of the locked loop is accomplished by modifying the control voltage applied to the PMOS and NMOS devices in the inverters. A 25 to 1 multiplexer is used to provide fine control for the QIE clocks. The system requirement is to be able to move the clocks in 2 nsec increments. Programmable registers allow for the selection of the correct delay stage, to set the QIE_CLK phase.

QIE_CLKa Mux

This block, programmed via the DLL Tap Select Register for QIEa, will select a properly phased QIE_CLKa from one of the 25 1ns steps available from the DLL. The selected clock signal will be driven as differential LVDS to QIEa.

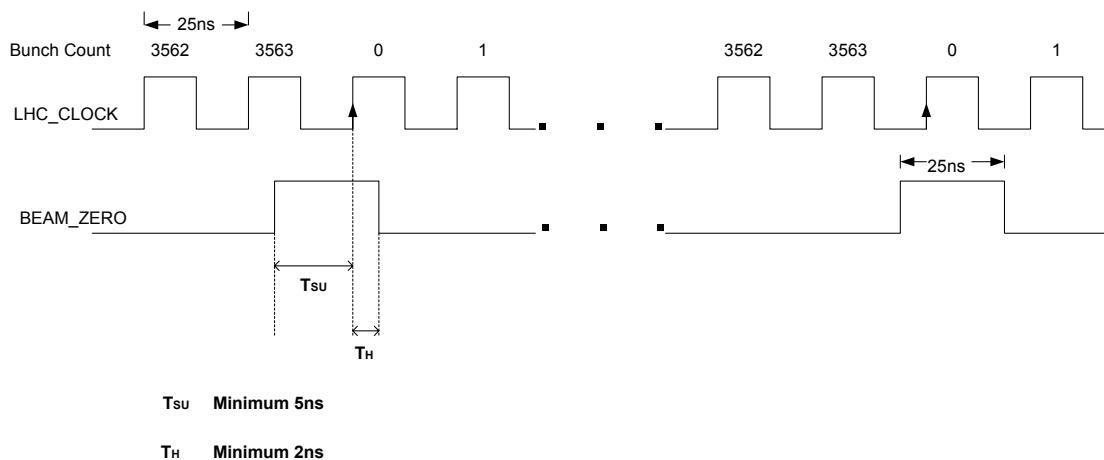
QIE_CLKb Mux

This block, programmed via the DLL Tap Select Register for QIEb, will select a properly phased QIE_CLKb from one of the 25 1ns steps available from the DLL. The selected clock signal will be driven as differential LVDS to QIEb.

Bunch Counter

This block counts the number of bunch crossings seen between beam zero markers. It increments with LHC_CLOCK and will be reset with each BC_ZERO, see figure 4. The final count prior to BC_ZERO must be latched and included in the Orbit Message. It must also be checked against the expected number to verify the correct functioning of the counter for each turn. If a mismatch (an error) is detected, a flag bit must be set and included in the Orbit Message.

The counter must be 12 bits wide.



The Bunch Count should be reset on rising edge of LHC_CLOCK whenever Beam Zero signal is present

Figure 4. Bunch Counter Operation

Test Pulse Trigger Comparator

This logic produces two Test Pulse Triggers which are triggered when the “Enable Test Pulse” bit is set in the Control Register and the Bunch Counter matches the Test Pulse Bunch Match Registers.

The Test Pulse produced should be 1 clock cycle long and in sync with the QIE clock pulse derived from the DLL and clock mux, see figure 5. The polarity of the Test Pulse is set through a bit in the Control Register.

Additionally, the Test Pulse must go through an alignment cycle similar to the QIE_RESET (see Appendix A). It will use the same mux settings as the QIE_RESET Pulse.

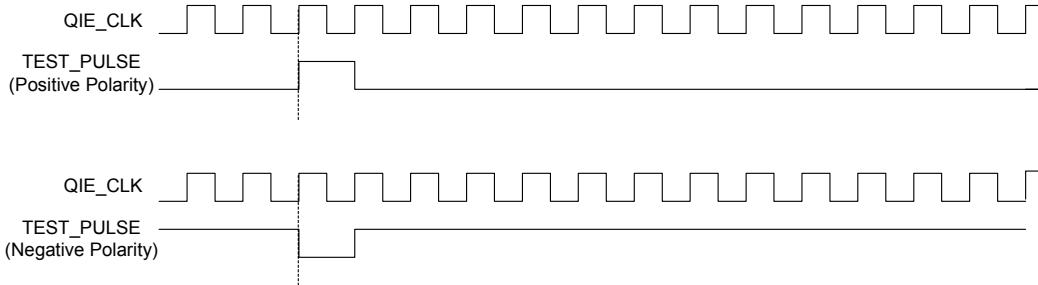


Figure 5. Test Pulse Timing

Event Synchronization Checker

This logic must verify that the final value of the Bunch counter, prior to reset by the BC_ZERO signal, matched the expected value. If it does not, the error condition must be flagged in the Orbit Message.

The Bunch counter should count from 0 to 3563 during normal operation.

CAP ID Check

The CAP ID Check block will compare the CapIDs which come from two QIE channels. CapIDs are expected to match if the QIEs have been properly reset and are receiving clocks. If the CapIDs from the two QIEs do not match, the CAP ID Check logic will flag a CapID error. The CapID error bit will be included in the Orbit Message.

QIE Data Alignment

This block will guarantee that the two channels of QIE are properly aligned.

Due to detector geometries and lengths of scintillating fibers, the worst case misalignment of data from the same event arriving at different QIEs will be 58 ns.

Appendix A presents a suggested logical implementation for the alignment block. The Alignment Control Registers for QIEa and QIEb are used to control the alignment features.

Synchronization

This block will guarantee that the QIE control signals, QIE_RESET and TEST_PULSE_TRIG, are properly timed as shown in figure 6.

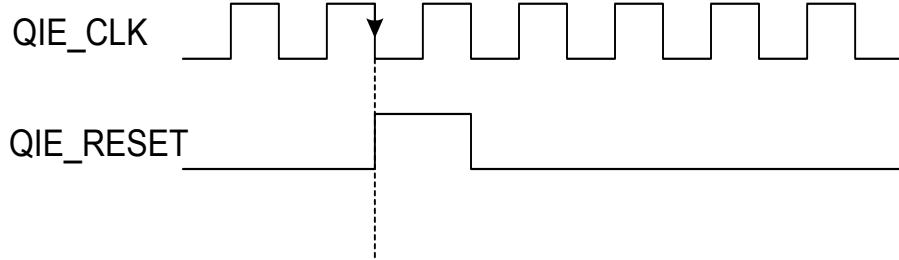


Figure 6. QIE_Reset Timing

Appendix A presents a suggested logical implementation for the alignment block. The Alignment Control Registers for QIEa and QIEb are used to control the alignment features.

Data Multiplexer

This block sends the data out to the gigabit serializing chip. It must choose to send either detector data, the Orbit Message, or a series of Fill Frames (to synchronise the link).

Data is clocked out of the CCA on the rising edge of the LHC Clock.

Orbit Message

The Orbit message should be sent whenever a Beam Zero is detected.

This block must control the assembling and sending of the Orbit Message. It receives inputs from the CAP ID checker, the Event Synchronization Checker and the Test Pattern Registers.

The source of data for the Orbit Message is controlled by the “Select Orbit Data” bit of the Control Register.

If the “Select Orbit Data” bit is low, the source of the Orbit Message will be the error information from the CAP ID checker and Bunch Counter synchronization checker, followed by fill frames.

If the “Select Orbit Data” bit is high, the source of the Orbit Message will be the Test Pattern Registers, followed by fill frames.

QIE Data Format

The CCA outputs data in the following format, unless the Orbit Message is being sent.

Data Out	Signal Name
Send Fill Frame	0
DATA	1
CNTRL	0
Data(18)	CAP_ID_Chан1(1)
Data(17)	CAP_ID_Chан1(0)
Data(16)	QIE_Chан1_Mant(0)
Data(15)	QIE_Chан1_Mant(1)
Data(14)	QIE_Chан1_Mant(2)
Data(13)	QIE_Chан1_Mant(3)
Data(12)	QIE_Chан1_Mant(4)
Data(11)	QIE_Exp1(0)
Data(10)	QIE_Exp1(1)
Data(9)	CAP_ID_Chан0(1)
Data(8)	CAP_ID_Chан0(0)
Data(7)	QIE_Chан0_Mant(0)
Data(6)	QIE_Chан0_Mant(1)
Data(5)	QIE_Chан0_Mant(2)
Data(4)	QIE_Chан0_Mant(3)
Data(3)	QIE_Chан0_Mant(4)
Data(2)	QIE_Exp0(0)
Data(1)	QIE_Exp0(1)

Orbit Message

If, “Select Orbit Data” bit of the Control Register = 0,

Data Out	Word 0 Event_Sync	Words 1-69 Fill Frames (See Note 1)
Send_Fill_Frame	0	1
DATA	0	X
CNTRL	1	X
Data(18)	X	X
Data(17)	X	X
Data(16)	X	X
Data(15)	X	X
Data(14)	BNCH_CNT Error	X
Data(13)	CapID Error	X
Data(12)	Bnch_Cnt(11)	X
Data(11)	Bnch_Cnt(10)	X
Data(10)	Bnch_Cnt(9)	X
Data(9)	Bnch_Cnt(8)	X
Data(8)	Bnch_Cnt(7)	X
Data(7)	Bnch_Cnt(6)	X
Data(6)	Bnch_Cnt(5)	X
Data(5)	Bnch_Cnt(4)	X
Data(4)	Bnch_Cnt(3)	X
Data(3)	Bnch_Cnt(2)	X
Data(2)	Bnch_Cnt(1)	X
Data(1)	Bnch_Cnt(0)	X

CAP ID Error = 0; no error occurred
 = 1; error detected

BNCH_CNT Error =0; no error occurred
 =1; error detected

Bnch_Cnt(11:0) is the number in the Bunch Counter prior to receiving the Beam_Zero Reset.

Or, if, “Select Orbit Data” bit of the Control Register = 1,
the 20 bytes of Test Pattern Data should be sent.

Word	SEND FILL FRAME	DATA	CNTRL	D18	D17	MSB(16:9)	LSB(8:1)
0	0	1	0	X	X	Test Pattern 1	Test Pattern 0
1	0	1	0	X	X	Test Pattern 3	Test Pattern 2
2	0	1	0	X	X	Test Pattern 5	Test Pattern 4
3	0	1	0	X	X	Test Pattern 7	Test Pattern 6
4	0	1	0	X	X	Test Pattern 9	Test Pattern 8
5	0	1	0	X	X	Test Pattern B	Test Pattern A
6	0	1	0	X	X	Test Pattern D	Test Pattern C
7	0	1	0	X	X	Test Pattern F	Test Pattern E
8	0	1	0	X	X	Test Pattern 11	Test Pattern 10
9	0	1	0	X	X	Test Pattern 13	Test Pattern 12
10-69	1	X	X	X	X	X	X

CCA Pinout Information
120 PIN quad Flat Pack – 14mm X 20mm
see Appendix B

Pin #	Pin Name	Description	Signal Type	Signal Level
1	N/C			
2	N/C			
3	N/C			
4	QIE_MANTb(0)	Bit(0) of QIE Mantissa – Channel_b (lsb)	Input	LVDS
5	QIE_MANTb(0)*	Bit(0)* of QIE Mantissa – Channel_b (lsb)	Input	LVDS
6	QIE_CAPIDb(0)	Bit(0) of QIE Capacitor ID – Channel_b (lsb)	Input	LVDS
7	QIE_CAPIDb(0)*	Bit(0)* of QIE Capacitor ID – Channel_b (lsb)	Input	LVDS
8	QIE_CAPIDb(1)	Bit(1) of QIE Capacitor ID – Channel_b (msb)	Input	LVDS
9	QIE_CAPIDb(1)*	Bit(1)* of QIE Capacitor ID – Channel_b (msb)	Input	LVDS
10	VDD		pwr/gnd	3.3V
11	GND		pwr/gnd	0V
12	QIE_CLKb	Phase adjusted differential clock provided to QIE – Channel_b	Output	LVDS
13	QIE_CLKb*	Phase adjusted differential clock provided to QIE – Channel_b	Output	LVDS
14	QIE_CLKa	Phase adjusted differential clock provided to QIE – Channel_a	Output	LVDS
15	QIE_CLKa*	Phase adjusted differential clock provided to QIE – Channel_a	Output	LVDS
16	LHC_CLK*		Input	LVPECL
17	LHC_CLK	40.08MHz clock	Input	LVPECL
18	GND		pwr/gnd	0V
19	QIE_RESETb	Reset signal provided to the QIEs once an orbit – Channel_b	Output	3.3V CMOS
20	QIE_RESETa	Reset signal provided to the QIEs once an orbit – Channel_a	Output	3.3V CMOS
21	TEST_PULSEb	Test Pulse produced by CCA which is sync'd with QIE_CLKb	Output	3.3V CMOS
22	TEST_PULSEa	Test Pulse produced by CCA which is sync'd with QIE_CLKa	Output	3.3V CMOS
23	CALIB_MODE	Forces QIE into Calibration Mode	Output	3.3V CMOS
24	QIE_RANGE(0)	Sets QIE Range (lsb) used when EN_QIE_RANGE is set high	Output	3.3V CMOS
25	QIE_RANGE(1)	Sets QIE Range (msb) used when EN_QIE_RANGE is set high	Output	3.3V CMOS
26	EN_QIE_RANGE	Forces QIE to use a given Range	Output	3.3V CMOS
27	PED_DACa(0)	Sets Pedestal DAC value driven to QIE – Channel_a (lsb)	Output	3.3V CMOS
28	PED_DACa(1)	Sets Pedestal DAC value driven to QIE – Channel_a	Output	3.3V CMOS
29	PED_DACa(2)	Sets Pedestal DAC value driven to QIE – Channel_a	Output	3.3V CMOS
30	PED_DACa(3)	Sets Pedestal DAC value driven to QIE – Channel_a (msb)	Output	3.3V CMOS

31	PED_DACb(0)	Sets Pedestal DAC value driven to QIE – Channel_b (lsb)	Output	3.3V CMOS
32	PED_DACb(1)	Sets Pedestal DAC value driven to QIE – Channel_b	Output	3.3V CMOS
33	PED_DACb(2)	Sets Pedestal DAC value driven to QIE – Channel_b	Output	3.3V CMOS
34	PED_DACb(3)	Sets Pedestal DAC value driven to QIE – Channel_b (msb)	Output	3.3V CMOS
35	DIEPDGND	Package leadframe die pad	gnd	0V
36	N/C			
37	N/C			
38	N/C			
39	N/C			
40	VDD		pwr/gnd	3.3V
41	VDDA	Analog power	pwr/gnd	3.3V
42	GNDA	Analog ground	pwr/gnd	0V
43	VSSD		pwr/gnd	0V
44	VDLL	Digital power	pwr/gnd	3.3V
45	VSSD		pwr/gnd	0V
46	GNDD	Ground	pwr/gnd	0V
47	RST_DLL*	DLL reset. Active low	Input	3.3V CMOS
48	RST_CCA*	Chip reset. Active low	Input	3.3V CMOS
49	BC_ZERO	BC ZERO Marker, comes once an orbit	Input	3.3V CMOS
50	RBX_A(2)	RBXbus Address line	Input	3.3V CMOS
51	RBX_A(3)	RBXbus Address line	Input	3.3V CMOS
52	RBX_A(4)	RBXbus Address line	Input	3.3V CMOS
53	RBX_A(5)	RBXbus Address line	Input	3.3V CMOS
54	RBX_A(6)	RBXbus Address line	Input	3.3V CMOS
55	RBX_A(7)	RBXbus Address line	Input	3.3V CMOS
56	RBX_CLK	RBXbus Serial Clock	Input	3.3V CMOS
57	VDLL	Digital power	pwr/gnd	3.3V
58	VSSD		pwr/gnd	0V
59	GNDD		pwr/gnd	0V
60	RBX_DATA	RBXbus Data	Input/O utput	3.3V CMOS - Open Drain
61	GND		pwr/gnd	0V
62	VSSD		pwr/gnd	0V
63	VDD		pwr/gnd	3.3V
64	N/C			
65	N/C			
66	N/C			
67	N/C			

68	N/C			
69	+2.5V	Digital power	pwr/gnd	2.5V CMOS
70	DATA	Data word flag sent to serializer	Output	2.5V CMOS
71	CNTRL	Control word flag sent to serializer	Output	2.5V CMOS
72	SEND_FF	Active high signal which enables sending of sync/idle words	Output	2.5V CMOS
73	TX_DATA(18)	Data sent to serializer	Output	2.5V CMOS
74	TX_DATA(17)	Data sent to serializer	Output	2.5V CMOS
75	TX_DATA(16)	Data sent to serializer	Output	2.5V CMOS
76	TX_DATA(15)	Data sent to serializer	Output	2.5V CMOS
77	TX_DATA(14)	Data sent to serializer	Output	2.5V CMOS
78	TX_DATA(13)	Data sent to serializer	Output	2.5V CMOS
79	TX_DATA(12)	Data sent to serializer	Output	2.5V CMOS
80	TX_DATA(11)	Data sent to serializer	Output	2.5V CMOS
81	TX_DATA(10)	Data sent to serializer	Output	2.5V CMOS
82	TX_DATA(9)	Data sent to serializer	Output	2.5V CMOS
83	TX_DATA(8)	Data sent to serializer	Output	2.5V CMOS
84	TX_DATA(7)	Data sent to serializer	Output	2.5V CMOS
85	TX_DATA(6)	Data sent to serializer	Output	2.5V CMOS
86	TX_DATA(5)	Data sent to serializer	Output	2.5V CMOS
87	TX_DATA(4)	Data sent to serializer	Output	2.5V CMOS
88	TX_DATA(3)	Data sent to serializer	Output	2.5V CMOS
89	TX_DATA(2)	Data sent to serializer	Output	2.5V CMOS
90	TX_DATA(1)	Data sent to serializer (lsb)	Output	2.5V CMOS
91	GND		pwr/gnd	0V
92	VSSD		pwr/gnd	0V
93	VDD		pwr/gnd	3.3V
94	QIE_EXPA(1)*	Bit(1)* of QIE exponent – Channel_a (msb)	Input	LVDS
95	QIE_EXPA(1)	Bit(1) of QIE exponent – Channel_a (msb)	Input	LVDS
96	QIE_EXPA(0)*	Bit(0)* of QIE exponent – Channel_a (lsb)	Input	LVDS
97	QIE_EXPA(0)	Bit(0) of QIE exponent – Channel_a (lsb)	Input	LVDS
98	QIE_MANTa(4)*	Bit(4)* of QIE Mantissa – Channel_a (msb)	Input	LVDS
99	QIE_MANTa(4)	Bit(4) of QIE Mantissa – Channel_a (msb)	Input	LVDS

100	N/C				
101	N/C				
102	N/C				
103	N/C				
104	QIE_MANTa(3)*	Bit(3)* of QIE Mantissa – Channel_a	Input	LVDS	
105	QIE_MANTa(3)	Bit(3) of QIE Mantissa – Channel_a	Input	LVDS	
106	QIE_MANTa(2)*	Bit(2)* of QIE Mantissa – Channel_a	Input	LVDS	
107	QIE_MANTa(2)	Bit(2) of QIE Mantissa – Channel_a	Input	LVDS	
108	QIE_MANTa(1)*	Bit(1)* of QIE Mantissa – Channel_a	Input	LVDS	
109	QIE_MANTa(1)	Bit(1) of QIE Mantissa – Channel_a	Input	LVDS	
110	QIE_MANTa(0)*	Bit(0)* of QIE Mantissa – Channel_a (lsb)	Input	LVDS	
111	QIE_MANTa(0)	Bit(0) of QIE Mantissa – Channel_a (lsb)	Input	LVDS	
112	QIE_CAPIDa(0)*	Bit(0)* of QIE Capacitor ID – Channel_a (lsb)	Input	LVDS	
113	QIE_CAPIDa(0)	Bit(0) of QIE Capacitor ID – Channel_a (lsb)	Input	LVDS	
114	QIE_CAPIDa(1)*	Bit(1)* of QIE Capacitor ID – Channel_a (msb)	Input	LVDS	
115	QIE_CAPIDa(1)	Bit(1) of QIE Capacitor ID – Channel_a (msb)	Input	LVDS	
116	QIE_EXPb(1)*	Bit(1)* of QIE exponent – Channel_b (msb)	Input	LVDS	
117	QIE_EXPb(1)	Bit(1) of QIE exponent – Channel_b (msb)	Input	LVDS	
118	QIE_EXPb(0)*	Bit(0)* of QIE exponent – Channel_b (lsb)	Input	LVDS	
119	QIE_EXPb(0)	Bit(0) of QIE exponent – Channel_b (lsb)	Input	LVDS	
120	QIE_MANTb(4)*	Bit(4)* of QIE Mantissa – Channel_b (msb)	Input	LVDS	
121	QIE_MANTb(4)	Bit(4) of QIE Mantissa – Channel_b (msb)	Input	LVDS	
122	QIE_MANTb(3)*	Bit(3)* of QIE Mantissa – Channel_b	Input	LVDS	
123	QIE_MANTb(3)	Bit(3) of QIE Mantissa – Channel_b	Input	LVDS	
124	QIE_MANTb(2)*	Bit(2)* of QIE Mantissa – Channel_b	Input	LVDS	
125	QIE_MANTb(2)	Bit(2) of QIE Mantissa – Channel_b	Input	LVDS	
126	QIE_MANTb(1)*	Bit(1)* of QIE Mantissa – Channel_b	Input	LVDS	
127	QIE_MANTb(1)	Bit(1) of QIE Mantissa – Channel_b	Input	LVDS	
128	N/C				

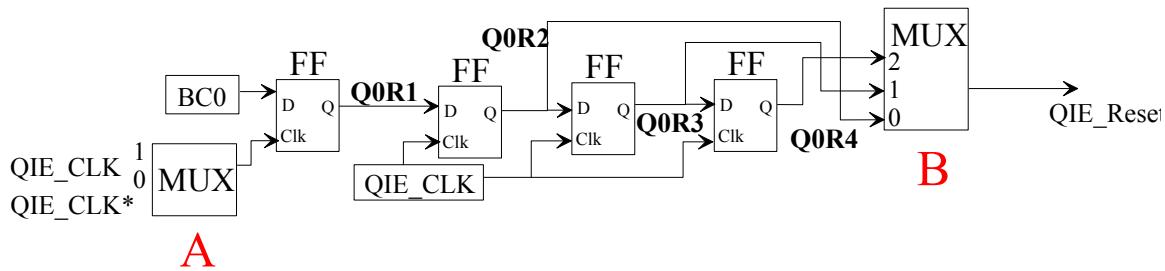
**LVDS like” signals will be generated by the QIE. These signals will not drive a cable, but will be sufficient to drive the couple of inches of traces we expect to use. The signal levels are compatible with LVDS levels and may be received and re-driven with a commercial LVDS receiver if desired.

APPENDIX A

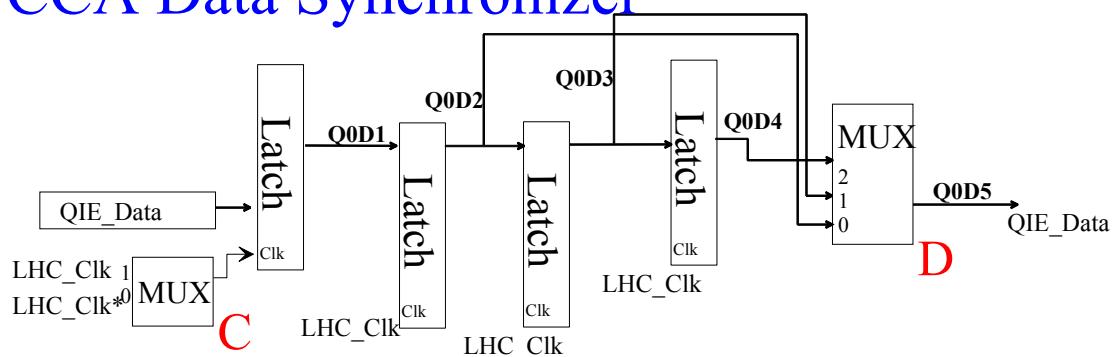
Channel Alignment

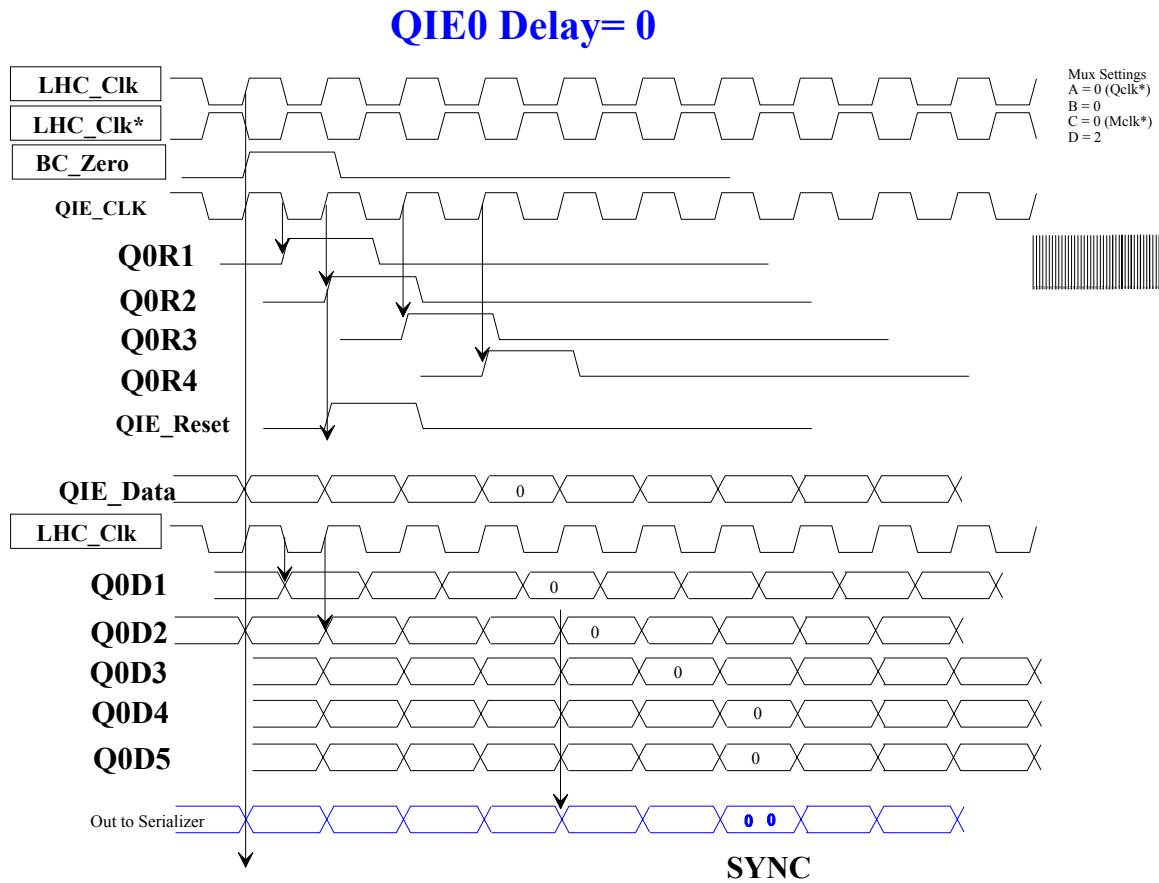
The following design and time settings are meant as a solution guide towards the problem of timing in QIE channels and resynchronizing the data to the LHC_CLK. The solution presented should allow QIE channels which are as much as 58ns out of sync, to be aligned properly and re-phased to the LHC_CLK.

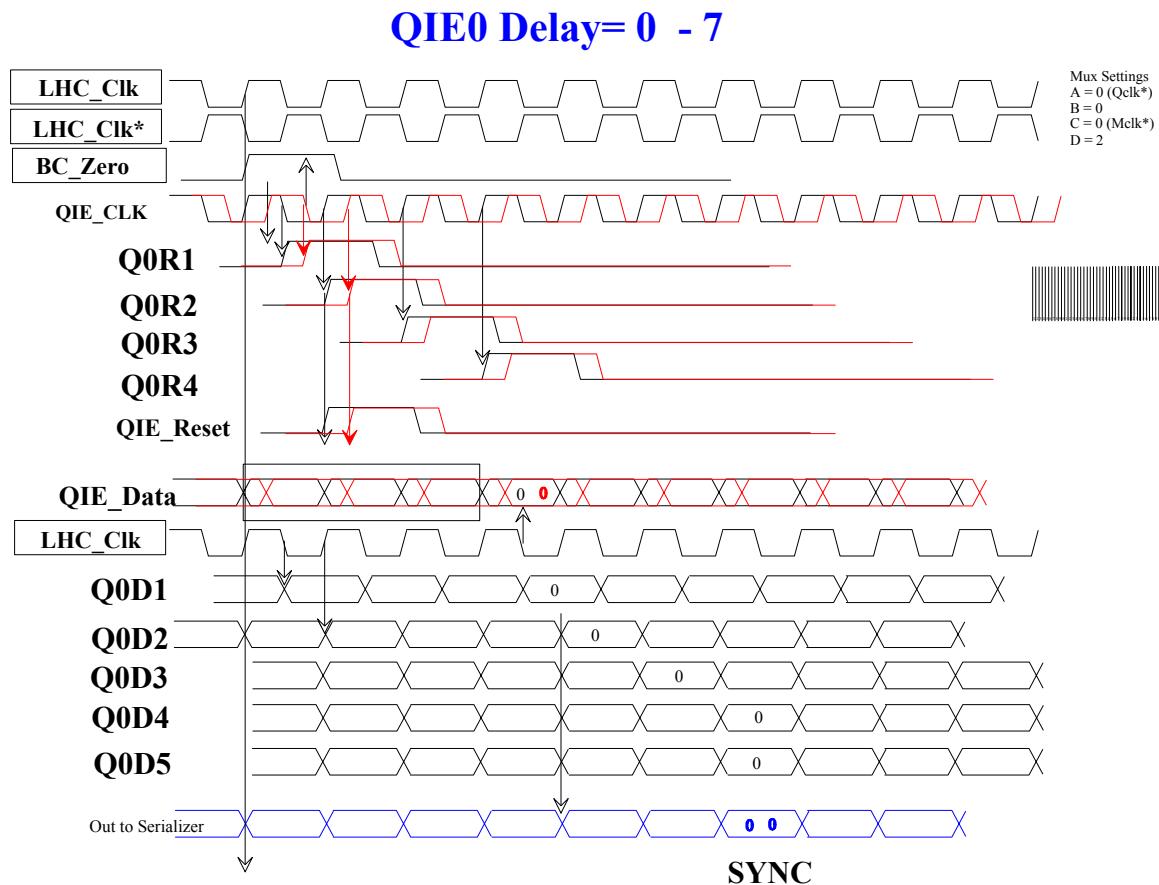
CCA Reset Synchronizer

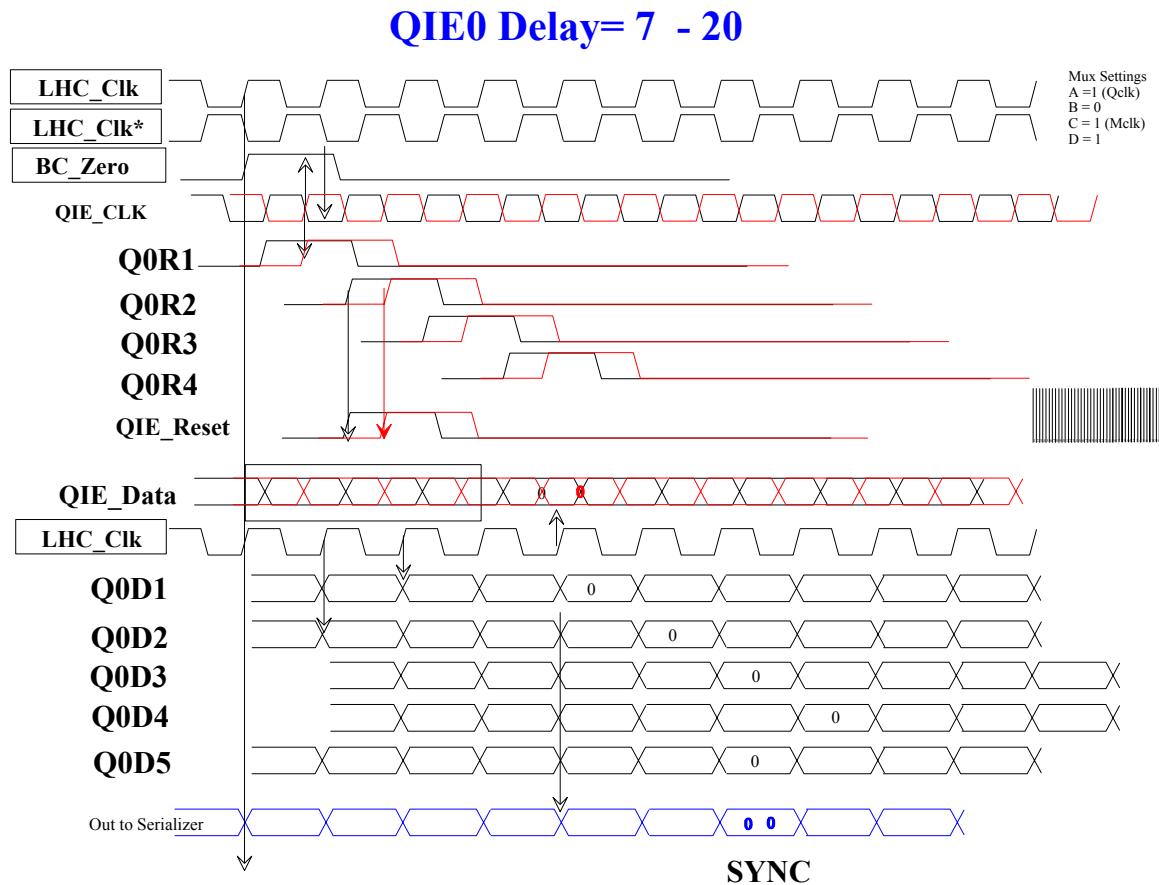


CCA Data Synchronizer

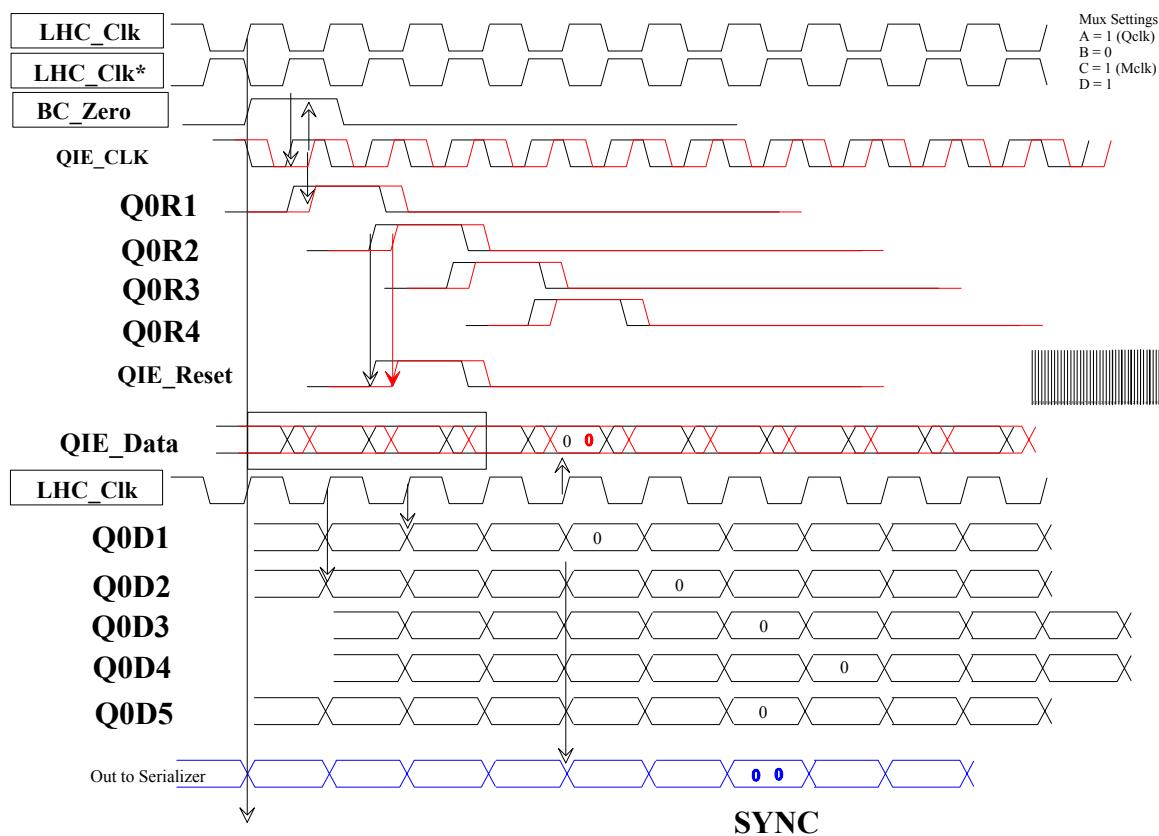




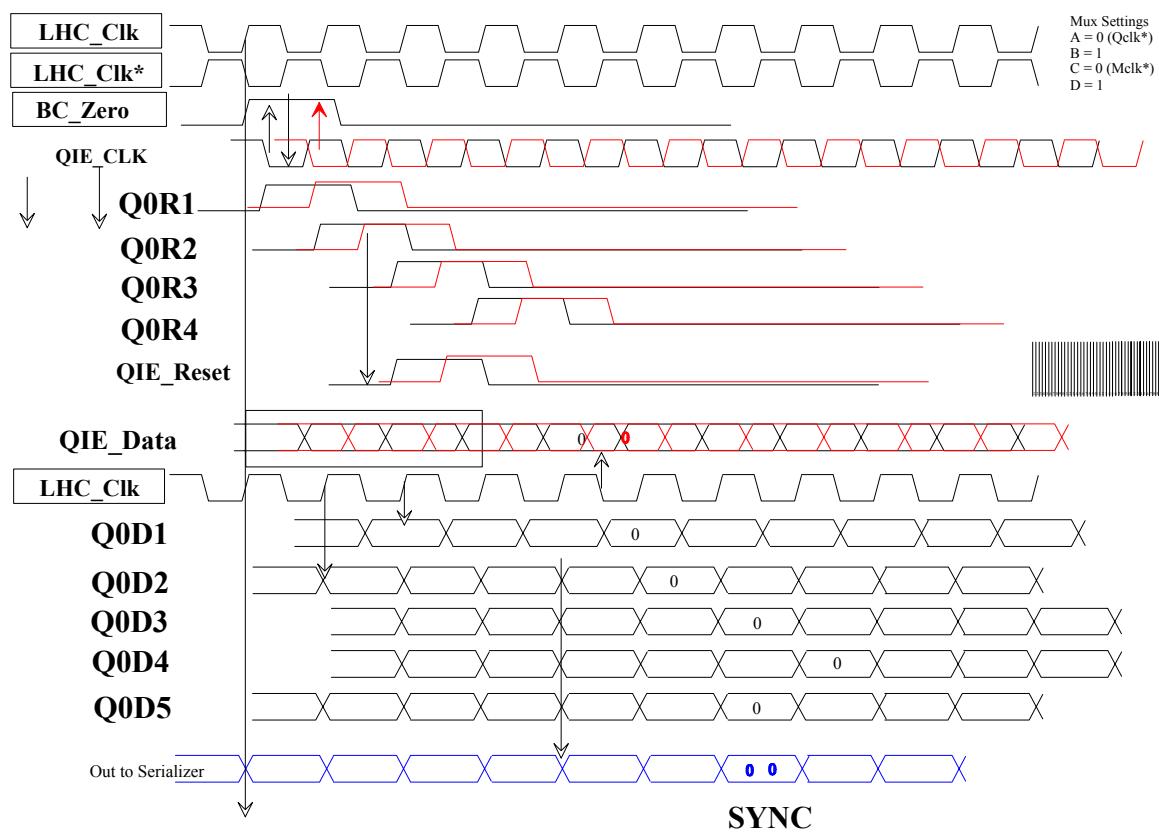




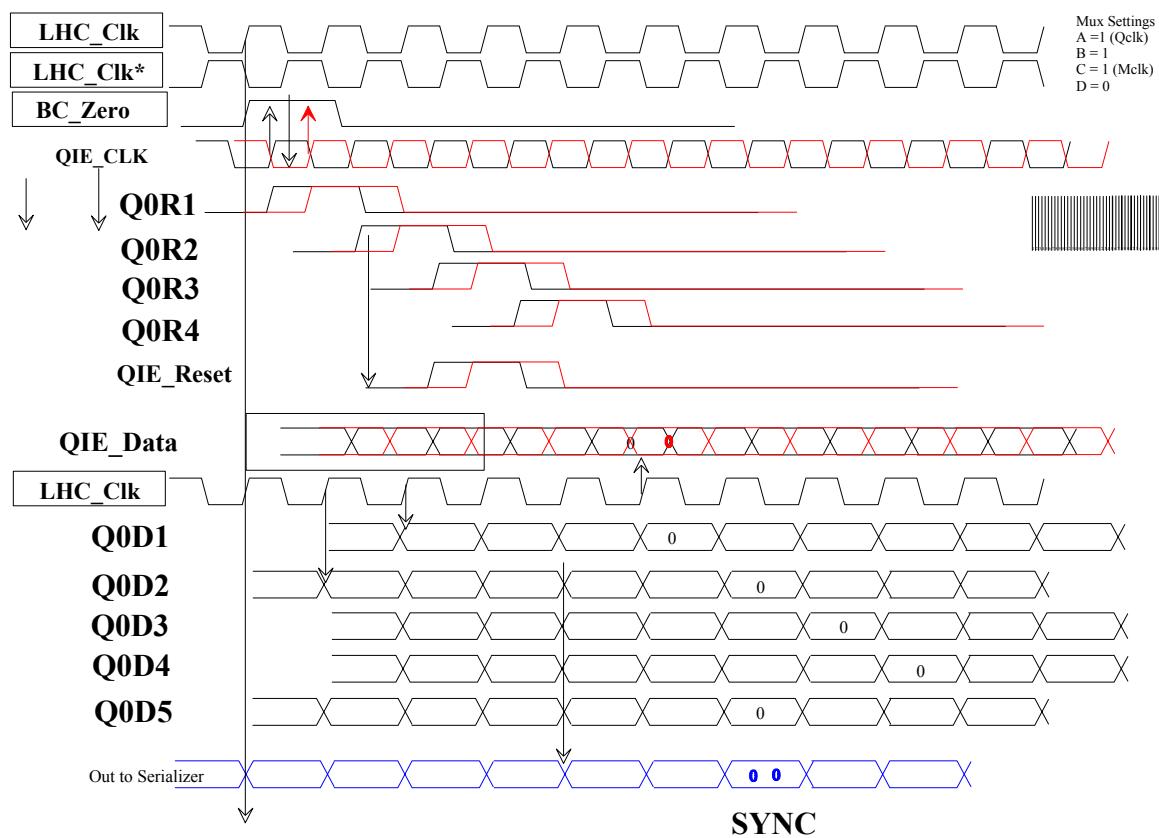
QIE0 Delay= 13-20



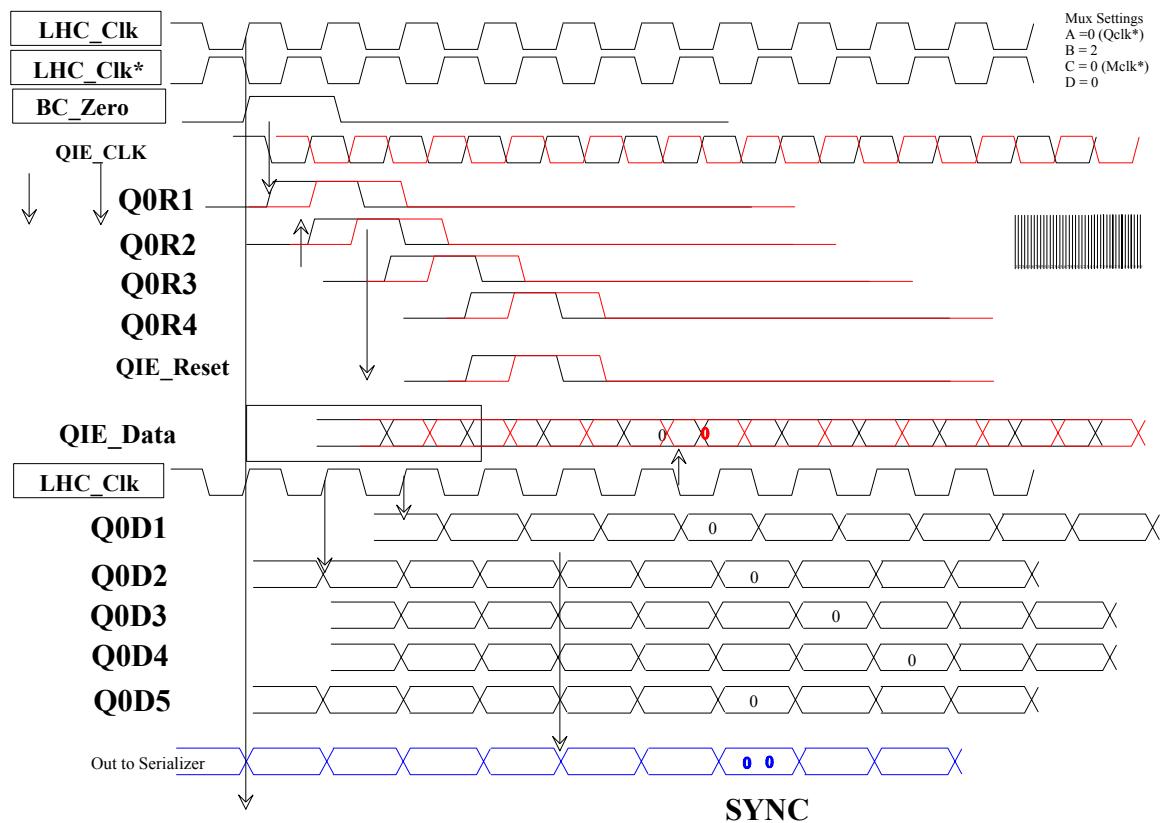
QIE0 Delay= 20-33



QIE0 Delay= 33 -45



QIE0 Delay= 45-58

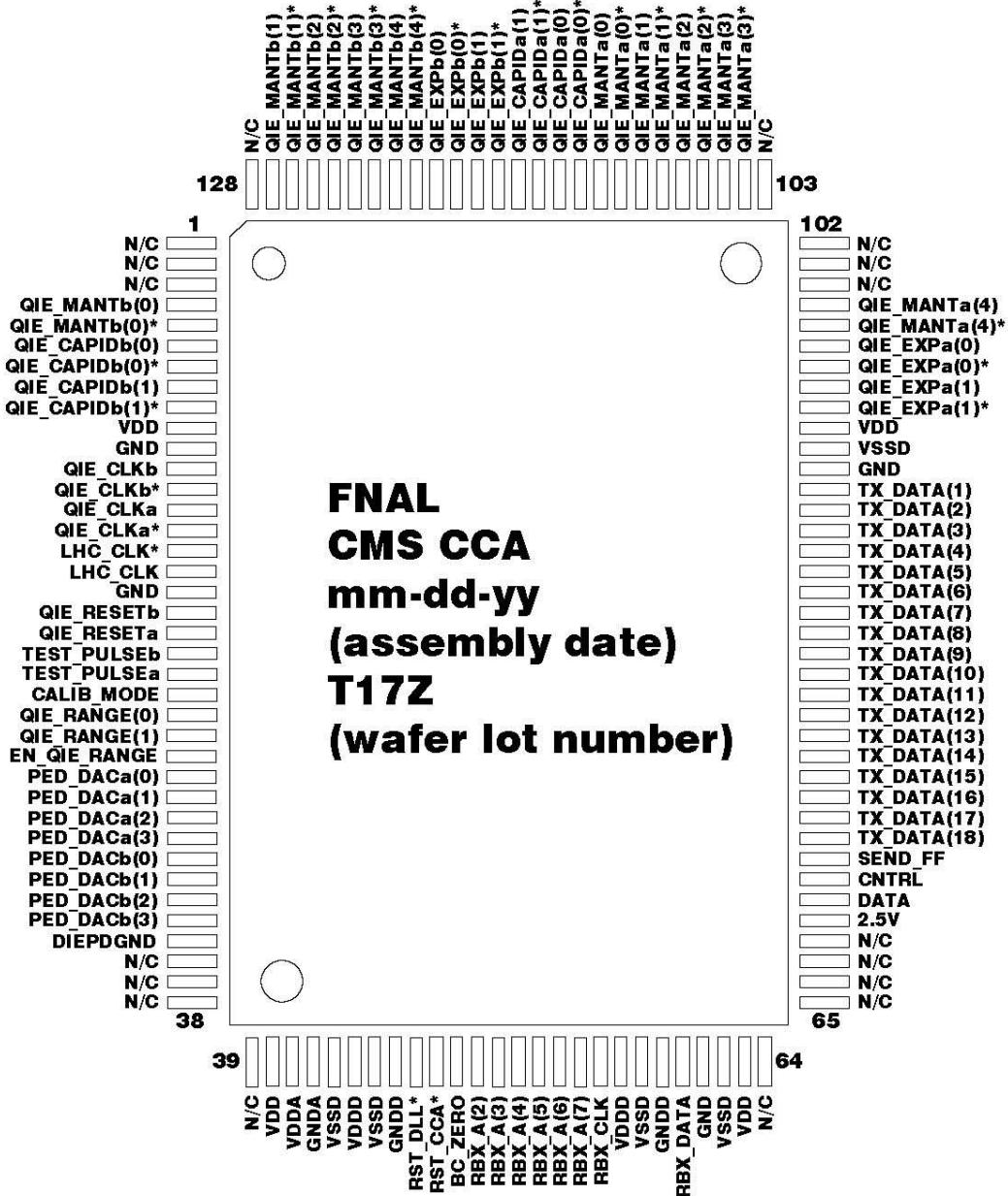


APPENDIX B

Pin/Pad Assignments Production Chip

CMS CCA

03/07/02

128-Lead QFP Package Pin Assignments

CMS CCA

03/07/02

128-Lead QFP Package Bonding Diagram